AMENDMENTS TO THE CLAIMS

What is claimed is:

1. (Original) A computer system comprising:

a main memory;

a cache;

a cache controller, which dynamically controls the cache such that a thread binary compiler divides a program into multiple threads and loads the program as a recompiled program whenever the cache loads the program from the main memory; and

a simultaneous multithreading processor, which processes instructions from threads such that the instructions from each thread can be executed simultaneously.

2. (Original) The computer system of claim 1, wherein the cache stores a plurality of recompiled programs generated from a plurality of programs.

3. (Original) The computer system of claim 1, wherein the recompiled program has a different instruction set architecture than the structure of the program loaded into the main memory.

4. (Original) The computer system of claim 1, wherein the recompilation is performed at an instruction level of the program.

5. (Original) The computer system of claim 1, wherein the thread binary compiler is software residing in the main memory.

6. (Original) The computer system of claim 1, wherein the thread binary compiler is loaded into and operates in the cache whenever the thread binary compiler recompiles the program.

7. (Original) The computer system of claim 1, wherein once the recompiled program has completed execution, the recompiled program is deleted.

8. (Original) The computer system of claim 1, wherein once the recompiled program has completed execution, the recompiled program is stored in the main memory.

9. (Original) The computer system of claim 8, wherein if the main memory is full, the recompiled program stored in the main memory is deleted.

10. (Original) The computer system of claim 8, wherein if the main memory is full, the recompiled program stored in the main memory is stored in the hard disk/flash memory.

11. (Original) A simultaneous multithreading method comprising;

loading a program into a main memory; loading a thread binary compiler to a cache;

dynamically controlling the cache such that the thread binary compiler divides the program into multiple threads and loads the program as a recompiled program, using the cache controller, whenever the cache loads the program from the main memory; and

processing instructions from threads such that the instructions from threads, using a simultaneous multithreading processor can be executed simultaneously.

12. (Original) The simultaneous multithreading method of claim 11, wherein the cache stores a plurality of recompiled programs generated from a plurality of programs.

13. (Original) The simultaneous multithreading method of claim 11, wherein the recompiled program has a different instruction set architecture than the structure of the program loaded into the main memory.

14. (Original) The simultaneous multithreading method of claim 11, wherein the recompilation is performed at an instruction level of the program.

15. (Original) The simultaneous multithreading method of claim 11, wherein the thread binary compiler is software residing in the main memory.

16. (Original) The simultaneous multithreading method of claim 11, wherein the thread binary compiler is loaded into and operates in the cache whenever the thread binary compiler recompiles the program.

17. (Original) The simultaneous multithreading method of claim 11, wherein once the recompiled program has completed execution, the recompiled program is deleted.

18. (Original) The simultaneous multithreading method of claim 11, wherein once the recompiled program has completed execution, the recompiled program is stored in the main memory.

19. (Original) The simultaneous multithreading method of claim 18, wherein if the main memory is full, the recompiled program stored in the main memory is deleted.

20. (Original) The simultaneous multithreading method of claim 18, wherein if the main memory is full, the recompiled program stored in the main memory is stored in the hard disk/flash memory.

21. (Original) A computer system comprising:

a main memory;

a cache;

cache controller means for dynamically controlling the cache such that a thread binary compiler divides a program into multiple threads and loads the program as a recompiled program whenever the cache loads the program from the main memory; and

a simultaneous multithreading processor, which processes instructions from threads such that the instructions from each thread can be executed simultaneously.

22. (Currently Amended) A cache controller system comprising:

a cache;

a main memory;

a cache controller wherein said cache~~n~~ controller dynamically controls the cache so that a thread binary compiler divides a program into multiple threads and loads the program as a recompiled program whenever the ~~cachen~~ cache loads the program from the main memory and forwards the threads to a simultaneous multithreading processor for processing.